

## METHOD FOR TRANSMITTING AND DOWNLOADING STREAMING DATA

### Field of the Invention

5

The present invention relates to a method for transmitting and downloading streaming data, more particularly to a method for transmitting and downloading streaming data through plurality of distributed nodes.

### 10 Background of the Invention

Conventionally, most contents data provided through the Internet were small size data such as document or picture. However, as data transmission technology and devices develop, multimedia contents are also widely provided. The  
15 multimedia contents are transmitted to user clients through streaming method.

Conventionally, streaming data were transmitted from one server to user clients.

The streaming data should be transmitted in real time. However, as the streaming data were transmitted from one server to user clients conventionally,  
20 transmission delay has occurred frequently when load of the server becomes higher or connection state is instable. Because the streaming data should be played as soon as they are downloaded, transmission delay affects QoS (Quality of Service) greatly unlike general file download.

As a prior art regarding file download, US patent 6,339,785 discloses a  
25 method for downloading files through multiple servers. According to the US

patent 6,339,785, total file is divided by N and file is downloaded through N servers so that file can be transmitted stably although error occurs in one connection.

However, according to the above file download method, the file is available only after whole data are downloaded. Therefore, the download method of US  
5 6,339,785 can not be applied to streaming data that should be played as soon as they are downloaded.

### Detailed Description of the Invention

10 In order to solve the above mentioned problems, the present invention intends to provide a method for transmitting and downloading streaming data through a plurality of nodes.

Another object of the present invention is to provide a method for downloading streaming data stably from plurality of user clients in P2P network and  
15 a communication agent program installed in user clients.

Another object of the present invention is to provide a method for downloading streaming data stably in file transmission system with multiple servers and a communication agent program installed in user clients.

Another object of the present invention is to provide a method for dividing  
20 streaming data into blocks and sub blocks and downloading data from plurality of connections.

In order to achieve above-mentioned objects, according to a preferred embodiment of the present invention, there is provided a method for downloading streaming data comprising the steps of: (a) establishing connections with a plurality  
25 of nodes; (b) sending a request for sub blocks of streaming data to the plurality of

nodes where connection is established to download the sub blocks; (c) monitoring download state of the established connections; and (d) redistributing sub blocks to be downloaded from some of the nodes where connection is established according to the monitoring result; wherein, the step (b) to step (d) are repeated for  
5 downloading sub blocks included in next block when all sub blocks included in a block are downloaded

According to the present invention, the step of determining sub blocks to download from each of the nodes where connection is established may be further comprised.

10 The step (c) may comprise the step of monitoring if there exists a connection where sub block download is completed.

The sub blocks to be downloaded from each of the nodes may be determined by calculating connection state valuation index.

The connection state valuation index may be calculated using information  
15 selected from group consisting of round-trip time with each of the nodes and average download speed from each of the nodes.

The step (d) may comprise the step of redistributing sub blocks between a connection where sub block download is completed and some of the connections where sub block download is not completed when a connection where sub block  
20 download is completed exists.

The step (d) may comprise the step of redistributing sub blocks between a connection where sub block download is completed and a connection of which download rate is the lowest among connections where sub block download is not completed.

25 The step for redistributing sub blocks between a connection where sub

block download is completed and a connection of which the download rate is the lowest may comprise the steps of: determining download speed of the connection where the sub block download is completed and the connection of which the download rate is the lowest; determining the number of remaining sub blocks to  
5 download in the connection of which the download rate is the lowest; determining if redistribution of sub blocks is necessary; redistributing the remaining sub blocks between the connection where sub block download is completed and the connection of which the download rate is the lowest according to rate of the download speed if sub block redistribution is necessary

10 According to another embodiment of the present invention, there is provided a communication agent program installed in a plurality user clients in the system where a connection control server and the plurality of user clients are connected through network, comprising: an agent manager module for providing information of contents stored in a user client and identification information of the  
15 user client to the connection control server and transmitting contents request information to the connection control server when a user requests contents; a node information manager module for receiving and storing list information of nodes that store requested contents; a connection control module for establishing connections with a plurality of nodes using the list information of nodes, and determining sub  
20 blocks to download from each of the nodes where connection is established to request sub blocks, and redistributing sub blocks to download from some of the connected nodes by monitoring download state while downloading sub blocks; a sub block manager module for determining if all sub blocks included in a block are downloaded and requesting sub blocks included in a next block to download sub  
25 blocks included in the next block when all sub blocks included in a block is

downloaded.

According to another embodiment of the present invention, there is provided a communication agent program installed in user clients downloading streaming data from system including a plurality of contents servers and at least a connection control server, comprising: a data request module for sending a request for transmission of data to the connection control server; a node information manager module for receiving and storing list of contents servers from which the requested data are to be downloaded; a connection control module for establishing connections with a plurality of contents servers included in the contents server list, and requesting sub blocks of streaming data to download from the contents servers where connection is established, and redistributing sub blocks to download in some of the contents servers where connection is established by monitoring download state while downloading sub blocks from connected contents servers; and a sub block manager module for determining if all sub blocks included in a block are downloaded and requesting sub blocks included in a next block to download sub blocks of the next block when all sub blocks included in a block are downloaded.

According to another embodiment of the present invention, there is provided a connection control server connected with a plurality of user clients through network for controlling connection between the user clients in order for a user client to download streaming data by connecting other user clients, comprising: a mesh manager module for receiving information of contents stored in the connected plurality of user clients and address information of the connected plurality of user clients, and providing information of nodes that store requested contents to a user client which requested contents; and a mesh information database for storing information of contents stored in each of the user clients and address

information of each of the user clients; wherein a communication agent program is installed in the plurality of user clients, the communication agent program controls user clients to establish connections with a plurality of nodes using the node information, and to send a request for sub blocks of streaming data to download to the connected nodes, and to redistribute sub blocks to download by monitoring download state of each of the connected nodes.

According to another embodiment of the present invention, there is provided a connection control server connected with a plurality of user clients and a plurality of contents servers through network for controlling connection between the user clients and the contents servers in order for a user client to download streaming data by connecting at least two contents servers, comprising: a server state determining module for receiving state information from the connected plurality of contents servers and determining state of each of the contents servers; a server list providing module for providing list information of servers to transmit streaming data of requested contents when receiving contents request information from a user client; wherein a communication agent program is installed in the user clients, the communication agent program controls the user clients to establish connections with at least two contents servers using the server list information, and determine sub blocks of streaming data to download from each of the connected contents servers to request sub blocks, and to redistribute sub blocks to download by monitoring download state of sub blocks from each of the connected nodes.

### **Brief Description of the Drawings**

FIG. 1 is a schematic configuration of streaming data transmission system

where the present invention is applied.

FIG. 2 is an example of network system where the method for transmitting streaming data according to the present invention is applied.

FIG. 3 is another example of network system where streaming data  
5 transmission method of the present invention is applied.

FIG. 4 is another example of network system where streaming data transmission method of the present invention is applied.

FIG. 5 is a structure of streaming data according to a preferred embodiment of the present invention.

10 FIG. 6 is a flow chart of streaming data download method according to a preferred embodiment of the present invention.

FIG. 7 is a flowchart of operation of the connection control server according to a preferred embodiment of the present invention.

FIG. 8 is a flow chart of operation of the connection control server  
15 according to another embodiment of the present invention.

FIG. 9 is a block diagram of agent program installed in the user client according to a preferred embodiment of the present invention.

FIG. 10 is a detailed block diagram of the connection control module according to a preferred embodiment of the present invention.

20 FIG. 11 is a flow chart of process that sub block is redistributed by determining connection state according to a preferred embodiment of the present invention.

FIG. 12 is a flow chart of process that is performed when there exists a connection where sub block download is completed according to a preferred  
25 embodiment of the present invention.

FIG. 13 is a block diagram of module of the connection control server according to a preferred embodiment of the present invention.

FIG. 14 is a block diagram of the module of the connection control server according to another embodiment of the present invention.

5

### **Best Mode for Carrying out the Invention**

The present invention will be described by virtue of the following embodiments in more detail.

10        FIG. 1 is a schematic configuration of streaming data transmission system where the present invention is applied.

Referring to FIG. 1, the streaming data transmission system according to the present invention may comprise a plurality of nodes 100, 102, 104, 106, 108, 110 and a user client 112.

15        The user client establishes connections with selected plurality of nodes 102, 106, 108 and downloads streaming data.

In FIG. 1, the plurality of nodes 100, 102, 104, 106, 108, 110 store contents data and transmit requested contents data to the user client after receiving contents request information from the user client. The node may be a server that generally  
20        transmits contents or other user client.

The user client 112 transmits contents request information to the plurality of nodes and downloads contents data from corresponding nodes 102, 106, 108. In other words, according to the present invention, streaming data are downloaded from plurality of paths through plurality of nodes.

25        At this time, the user client sends a request for different streaming data to



each of the nodes where connection is established and assembles the streaming data downloaded from the plurality of nodes.

FIG. 5 is a structure of streaming data according to a preferred embodiment of the present invention.

5 Referring to FIG. 5, the streaming data according to the present invention comprise a plurality of blocks 500 and each of the blocks comprises a plurality of sub blocks 502.

The block and sub block are logical unit for classifying streaming data and it would be obvious to those skilled in the art that other terms can be used.

10 Unlike general file download, the streaming data should be played while being downloaded. Therefore, general file download method through multiple servers cannot be applied to the streaming data download because conventionally whole file data are divided into N segments and each of the segments is downloaded through multiple paths. According to the conventional file download method  
15 using multiple paths, downloaded file cannot be played until whole file data are downloaded while streaming data should be played as soon as they are downloaded.

Therefore, according to the present invention, whole streaming data are divided into blocks and sub blocks included in a block are downloaded in parallel/distribution method through multi-paths.

20 For example, in the system of FIG. 1, if one block comprises ten sub blocks, the user client 112 requests 1~3 sub blocks to the second node 102, 4~7 sub blocks to the fourth node 106 and 8~10 sub blocks to the fifth node 108.

If all sub blocks included in a block are downloaded, the user client 112 requests sub blocks included in the next block. After all data of a block are  
25 downloaded, sub blocks of the next block are downloaded, and therefore data can

be played while being downloaded although data are downloaded in multi-paths.

In FIG. 5, the case where the streaming data are divided into blocks and sub blocks, however, it would be obvious to those skilled in the art that the streaming data can be divided into smaller units than the sub block and can be transmitted by the smaller units.

When the user client sends a request for sub blocks to each of the nodes, the user client determines the connection state of each node. For example, among the second node 102, the fourth node 106 and the fifth node 108, if the connection state of the second node 102 is the best and the connection state of the fifth node is the worst, the user client 112 sends a request for the maximum number of sub blocks to the second node and minimum number of sub blocks to the fifth node. The method for determining the state of the node and requesting sub blocks according to the result of determination will be explained in more detail referring to another figure.

When sub blocks are downloaded through the multi-paths, sub block download in a connection can be completed earlier than in other connections, and in this case, data are not transmitted in the connection where sub block download is completed until sub block download is completed in other connections. Idle connection of this type reduces transmission efficiency.

Therefore, according to the present invention, if sub block download is completed in a connection among the plurality of connections, some sub blocks which are not transmitted are downloaded through the connection where the sub block download is completed. The method for redistributing sub blocks to be downloaded when a sub block download is completed in a connection will be explained in more detail referring to another figure.

FIG. 2 is an example of network system where the method for transmitting

streaming data according to the present invention is applied.

FIG. 2 illustrates a case where streaming data transmission method using P2P (Peer to Peer) is applied.

Referring to FIG. 2, the network system where the present invention is applied may comprise a connection control server 200 and a plurality of clients 202, 204, 206, 208, 210, 212, 214, and 216.

The connection control server 200 receives contents request information from a user client and provides address information of other user clients that store the requested contents to the user client that requested contents in response to the contents request information. The user client that requested contents downloads streaming data of contents through multi-paths using the provided address information.

For example, if a user client 210 requests movie 'A' to the connection control server 200, the connection control server 200 derives address information of user clients that store movie 'A' and provides the address information to the user client 210 that requested the contents.

The address information may be IP address and port number and it would be obvious to those skilled in the art that other address identification information can be used.

For example, let us assume that the user clients that store the movie 'A' are reference number 202, 204 and 208 in FIG. 2. If the connection control server provides address information of clients 201, 204, 208 to the user client 210 that requested contents, the user client 210 tries connection establishment with the user clients 202, 204, and 208.

If the connection is established, the user client 210 requests different sub

blocks to each of the clients 202, 204, 208 where connection is established. If all sub blocks included in a block are downloaded, the user client 210 requests sub blocks included in the next block. The transmitted streaming data are stored in the user client. If other clients request the streaming data, the stored streaming data  
5 are transmitted.

A communication agent program for communication with the connection control server or other clients is installed in each of the user clients. In the embodiment of the present invention, the case where the communication agent is computer software is described, however, it would be apparent to those skilled in  
10 the art that the communication agent may be an independent device or a card.

If power of a client where the communication agent is installed is on or the communication agent is executed, the agent program transmits information of contents stored in the user client to the connection control server 200 and the connection control server 200 stores the transmitted information.

15 ~~FIG. 3 is~~ FIG. 3 is another example of network system where streaming data transmission method of the present invention is applied.

FIG. 3 illustrates a case where streaming data are provided through multiple servers.

Referring to FIG. 3, the streaming data transmission system according to an  
20 embodiment of the present invention may comprise a plurality of contents servers 300, 302, 304, 306, 308, a connection control server 310 and a user client 312.

The connection control server 310 receives contents request information from the user client 312 and provides information of servers where the user client should be connected among the plurality of servers 300, 302, 304, 306, 308.

25 The connection control server 310 receives state information from the

plurality of servers 300, 302, 304, 306, 308 periodically and controls the user client to be connected to the servers with good state.

The user client 312 receives server list information from the connection control server 310 and establishes connection with the contents servers in the list to  
5 download streaming data through multiple paths.

As in the case of FIG. 2, a communication agent program is installed in the user client for downloading streaming data through multiple paths.

FIG. 4 is another example of network system where streaming data transmission method of the present invention is applied.

10 FIG. 4 is a system where P2P (Peer to Peer) system of FIG.2 and multiple server system of FIG. 3 are combined.

Referring to FIG. 4, the streaming data transmission system according to a preferred embodiment of the present invention may comprise a connection control server 410, a plurality of contents servers 400, 402, 404, 406, 408 and a plurality of  
15 clients 412, 414, 416, 418, 420, 422, 424, 426, 428. Although the plurality of contents servers are illustrated in FIG. 4, the number of contents server can be one unlike FIG. 4.

In FIG. 4, the connection control server 410 receives contents request information from user clients, and provides list information of servers or user  
20 clients that store the requested contents.

Like FIG. 2 and FIG. 3, an agent program is installed in the user clients 412, 414, 416, 418, 420, 422, 424, 426, 428 and the agent program transmits contents information stored in the client to the connection control server 410 when power of the client is on or the agent program is executed.

25 The user client that requested contents establishes connection with contents

servers or clients using the list information that is provided by the connection control server 410 and sends a request for different sub blocks to the clients or contents servers where connection is established to download streaming data.

In case of FIG. 4, the node list provided by the connection control server 410 can include user clients and server together. For example, if contents data are received from 5 nodes, one node may be a contents server and other 4 nodes are user clients. In this case, according to a preferred embodiment of the present invention, in order to reduce load of the contents server, contents server can be excluded from nodes if download state from user clients besides contents server is good. Further, the number of sub blocks requested to the contents server can be reduced.

In FIG.2 to FIG. 4, systems which can download streaming data through multiple paths are described. It would be obvious to those skilled in the art that others systems which transmit data through plurality of nodes besides systems of FIG.2 to FIG. 4 can be applied to the present invention.

FIG. 6 is a flow chart of streaming data download method according to a preferred embodiment of the present invention.

Referring to FIG. 6, a user client receives node list information from the connection control server S600. In system of FIG. 2, nodes are user clients that stores contents. In system of FIG. 3, nodes are plurality of contents servers, and in system of FIG. 4, nodes can include both of user clients and contents server.

If node list information is received, the user client establishes connections with some or all of nodes included in the node list S602. According to a preferred embodiment of the present invention, the user client establishes TCP connection, and other connection establishment method can be used. If connection

establishment fails, information of nodes where connection establishment failed is stored in a black list queue, and connection establishment is not tried for nodes stored in the black list queue in following connection establishment.

If connection is established, the user client determines connection state with  
5 nodes and determines sub block number to be downloaded from the nodes S604.

According to an embodiment of the present invention, the connection state is determined by round-trip time with each node or download speed from each node.

According to another embodiment of the present invention, sub blocks to be downloaded can be determined without connection state information in initial state  
10 of download. For example, if streaming data are downloaded from 3 nodes and one block comprises 15 sub blocks, the user clients requests 5 sub blocks for each of the 3 nodes.

If sub blocks to be downloaded from each of the nodes are determined, the user clients send a request for transmission of sub blocks to the nodes where  
15 connection is established S606. The user client may send a request for sub blocks to all nodes where connection is established or to some of the nodes where connection is established.

The user client downloads sub blocks from nodes where connection is established S608. Although it is not shown in FIG. 6, the user client determines if  
20 download error occurs while receiving sub blocks. According to a preferred embodiment of the present invention, occurrence of download error is determined by checking checksum value of sub blocks.

Alternatively, download error can also be determined by checking the checksum value of block after receiving all blocks.

25 While receiving sub blocks, the user client determines if a connection where

all requested sub blocks are downloaded exists S610.

If there exists a connection where all requested sub blocks are downloaded, the user client redistributes sub blocks of which the download is not completed to the connection where sub block download is completed to download redistributed  
5 sub blocks from the connection so that the connection where the sub block download is completed does not become idle state.

The user client determines if all sub blocks included in a block are downloaded S614.

If all sub blocks are downloaded, the user client repeats the process that  
10 determines numbers of sub blocks in the next block to be downloaded from each of the nodes through multiple paths S604.

The steps from S610 to S614 are repeated until all sub blocks included in a block are received.

As described referring to FIG. 6, according to the present invention, if sub  
15 block download is completed in a path(=connection), sub blocks to be downloaded from each of the nodes are redistributed in order to avoid occurrence of idle connection, by which data download efficiency can be improved.

In FIG. 6, a case that sub blocks are redistributed for the nodes that are transmitting streaming data to the user client. However, it would be obvious to  
20 those skilled in the art that sub blocks can also be redistributed for the nodes that are not transmitting sub blocks by monitoring the state of nodes to which sub block transmission is not requested.

Further, although it is not shown in FIG. 6, the download mode can be changed so that streaming data are downloaded from a singular server as  
25 conventional method if streaming data cannot be downloaded from the plurality of



nodes.

FIG. 7 is a flowchart of operation of the connection control server according to a preferred embodiment of the present invention.

FIG. 7 illustrates operation of the connection control server in P2P network  
5 of FIG. 2 or network where P2P and multiple servers are combined.

Referring to FIG. 7, the connection control server receives address information and contents information from the connected nodes S700. As described above, the communication agent program is installed in each of the user client (node) and the communication agent program transmits information of  
10 contents stored in the user client and information of address of the user client to the connection control server. The received node address information and node contents information are stored in the database of the connection control server.

The connection control server receives contents request information from connected nodes S702. Using the information received in step S700, the  
15 connection control server that received contents request information searches node that stores the requested contents S704.

The connection control server provides node list information to the user client that requested contents data S706.

Although it is not shown in FIG. 7, the connection control server may also  
20 receive node state information (for example, PING information) and may provide received node state information together with the node list information to the user client.

After receiving node list information, the user client selects node where connection is to be established using the state information. If connection  
25 establishment fails or data download rate is not good, the user client selects new

nodes where connection is to be established using the node state information. The node state information can be updated continuously, the updated information is provided to the user client which requested contents data or is downloading data.

FIG. 8 is a flow chart of operation of the connection control server according to another embodiment of the present invention.

FIG. 8 illustrates operation of the connection control server network using multiple servers, such as FIG. 3.

Referring to FIG. 8, the connection control server receives state information from each of the contents servers S800. According to a preferred embodiment of the present invention, the state information may be one or combination selected from group comprising bandwidth information, CPU usage rate information, memory usage rate information, the number of users connected, and file I/O information. It would be obvious to those skilled in the art that other information besides above-described information can also be used as the state information of server.

The connection control server receives contents request information from a user client S802.

The connection control server that received contents request information provides address information of servers with good state (in other words, node list information) using the state information in S800 and sub block distribution information (information on which sub blocks to be downloaded from each of the contents servers) to the client that requested contents S804.

In FIG. 8, the case that sub block distribution information is transmitted to the user client from the connection control server, however, it would be obvious to those skilled in the art that the client itself can distribute sub blocks to download by

checking the connection state after connecting the contents servers.

According to another embodiment of the present invention, in initial state of data download, the connection control server provides sub block distribution information. Then, if download of a block data is completed, the user client itself  
5 distributes sub blocks to download from each of the contents servers in the download of the next sub block because contents server can determine download speed with each of the contents servers.

FIG. 9 is a block diagram of agent program installed in the user client according to a preferred embodiment of the present invention.

10 Referring to FIG. 9, the communication agent program according to a preferred embodiment of the present invention may comprise an agent manager module 900, a cache manager module 902, a connection control module 904, a node manager module 906, a block manager module 908 and a sub block manager module 910.

15 In FIG. 9, the agent manager module 900 performs communication with the connection control server and provides user client information including IP address and port number and information of contents stored in the user client to the connection control server when the power of the user client is on or the agent program is executed.

20 Further, the agent manager module 900 provides contents request information to the connection control server when user requests contents. The agent manager module may provide meta information of the contents to a player which plays contents data when meta information is necessary for play of the requested contents.

25 The cache manager module 902 manages streaming data downloaded from

nodes. The cache manager module 902 provides information for remaining capacity of the cache and information of contents stored in the cache. The cache manager module 902 provides information of contents stored in the cache when power of the user client is on or the agent program is executed. According to a preferred embodiment of the present invention, the cache manager module 902  
5 scrambles the downloaded data when the downloaded data are stored.

The connection control module 904 establishes connections with predetermined number of nodes using the node list information received from the connection control server. The connection control module 904 distributes sub  
10 blocks to download from each of the nodes by determining the connection state of nodes and requests sub blocks to each of the nodes according to the distribution result.

Further, the connection control module 904 monitors the download states with each of the nodes continuously, and redistributes sub blocks to download from  
15 each of the nodes according to the monitoring result.

According to a preferred embodiment of the present invention, the connection control module 904 monitors if there exists connection where sub blocks download is completed, and the connection control module 904 redistributes sub blocks between the connection where sub block download is completed and some of  
20 the connections where sub block download is not completed to avoid idle connection.

According to another embodiment of the present invention, sub blocks can be redistributed previously according to the download state with each of the nodes although the connection where the sub block receipt is not completed does not exist.

25 According to another embodiment of the present invention, the connection

control module monitors state information of nodes where connection is not established, and redistributes sub blocks between some of the nodes that are transmitting data and some of the nodes that are not transmitting data.

As mentioned above, in the system of FIG. 4, the connection control module  
5 requests data to both of the server and the user clients, and if the download state from the user clients is good, the server may be excluded from nodes or small amount of data may be requested to the server in order to reduce load of the server.

The node manager module 906 provides node list information provided from the connection control server to the connection control module 904. As described  
10 above, the node list information may include IP address and port number. According to a preferred embodiment of the present invention, the agent manager module 900 requests information of nodes that store the requested contents and provides the node list information to the node manager module 906. The node information is updated continuously because power of nodes may be off or  
15 connection state may become instable.

The sub block manager module 910 downloads sub blocks from nodes and determines if all sub blocks included in a block are downloaded. If all sub blocks are downloaded, the sub block manager module 910 provides downloaded block data to the block manager module 908. If the sub block manager module  
20 determines all sub blocks included in a block are downloaded, the connection control module requests sub blocks of the next block.

The block manager module 908 provides received block data to a player which plays streaming data or to the cache manager module.

Although it is not shown in FIG. 9, a reservation manager module can be  
25 further included in the communication agent program. When streaming data are

transmitted with high speed, there may occur a case that the client cannot download data at predetermined speed on account of instability of internet connection state. At this case, the movie data are pre-stored in the cache through the reservation manager module before playing of the movie data and the reservation manager  
5 module performs processes for the reservation.

The reservation manager module manages information on streaming data which a user requested for reservation and monitors if the reserved contents are received. The reservation manager module requests data through the connection control module until the download of requested data is completed.

10 FIG. 10 is a detailed block diagram of the connection control module according to a preferred embodiment of the present invention.

Referring to FIG. 10, the connection control module may comprise a connection establishment module 1000, a connection state determining module 1002, a sub block distribution module 1004 and a download state monitoring  
15 module 1006.

The connection establishment module 1000 establishes connections with nodes from which data are to be downloaded using the node list information provided from the connection control server. The connection establishment may comprise following steps;

- 20 (i) A user client sends a request for TCP connection to a node.  
(ii) The node determines if the TCP connection can be allowed.  
(iii) The node allows TCP connection.

As explained above, other connection establishment methods other than TCP can be employed.

25 The connection state determining module 1002 determines the connection

state with each of the nodes. The connection state determining module 1002 calculates connection state valuation index for determining connection state.

According to an embodiment of the present invention, the connection state valuation index may be calculated using round-trip time with each of the nodes.

5        According to another embodiment of the present invention, the connection state valuation index can be calculated using download speed information from each of the nodes. Alternatively, both of the round-trip time and download speed can be used for calculation of connection state valuation index.

10       It is desirable that the connection state valuation index is calculated using the round-trip time initially, because download speed is not known initially, and the connection state valuation index is calculated using the download speed after data download is performed. As the download speed changes continuously, the connection state valuation index is also updated depending on the change of download speed.

15       According to another embodiment of the present invention, the connection state determining module receives information on state of servers from the servers and determines the connection state using the state information.

20       The sub block distribution module 1004 distributes sub blocks using the connection state valuation index calculated by the connection state determining module 1002. It is preferable that the sub block distribution module 1004 determines the number of sub blocks to download from each of the nodes according to the ratio of connection state valuation index.

25       The download state monitoring module 1006 determines if there exists a connection where sub block download is completed and requests sub block redistribution to the sub block manager module 1004 when a connection where sub

block download is completed exists.

It is not efficient if sub blocks are redistributed for all connected nodes, because control signal should be transmitted to the all nodes. Therefore, according to a preferred embodiment of the present invention, sub blocks are redistributed  
5 between a connection where sub block download is completed and a connection of which the download rate is the lowest. At this case, control signal is transmitted to only the node which completed the sub block transmission and the node of which data transmission rate is the lowest.

FIG. 11 is a flow chart of process that sub block is redistributed by  
10 determining connection state according to a preferred embodiment of the present invention.

Referring to FIG. 11, round-trip times with each of the connected nodes are measured in initial state of download S1100. Round-trip time means time duration between transmitting check signal to a connected node and receiving the response  
15 signal of the check signal.

After measuring round-trip signal, it is determined if average download speed can be known in all connections S1102. If several blocks are downloaded, average download speed in all connections can be known.

When average download speed in all connections can be known, the  
20 connection state valuation index is calculated using the average download speed information S1104. For example, when streaming data are downloaded from 3 nodes and average download speed of 3 nodes is 500,000bps, 100,000bps and 64,000bps, the connection state valuation index in each connection is 500,000, 100,000 and 64,000 respectively. Normalized value of average download speed  
25 can also be used as connection state valuation index.



When average download speed in all connections cannot be known, it is determined if average download speed in some of the connections can be known S1106. This case can occur when streaming data are downloaded from new node while downloading streaming data.

- 5           When average download speed can be known in some of the connections, the connection state valuation index is calculated using round-trip time and average download speed S1108.

For example, assuming that streaming data are downloaded from 3 nodes, and average download speed of the first node is 200,000bps, average download  
10   speed of the second node is 100,000bps and average download speed of the third node is not known. Further, assuming that round-trip time of the first node is 23ms, round-trip time of the second node is 32ms and round trip time of the third node is 20ms.

In this case, the connection state valuation index is calculated by (download  
15   speed)/(round-trip time). Average download speed of the third node is predicted using the value of which the (download speed)/(round-trip time) is the smallest. As the (download speed)/(round-trip time) of the second node is the smallest, the average download speed of the third node is calculated by the following equation 1.

[Equation 1]

20

$$\frac{100,000}{20} \times 32 = 160,500$$

If average download speed in all connections cannot be known, the connection state valuation index is calculated using round-trip time. According to

an embodiment of the present invention, reciprocal of the round-trip time can be used as the connection state valuation index.

For example, assuming that streaming data are downloaded from 3 nodes and round-trip time of each node is 23ms, 41ms and 32ms. In this case, the  
5 connection state valuation index of each node may be 1/23, 1/42 and 1/32 respectively.

After calculation of connection state valuation index, the number of sub block to be downloaded from each of the nodes among all sub blocks is determined.

The number of sub block to be downloaded from each of the nodes  
10 determined by the connection state valuation index. In more detail, the number of sub block to be downloaded is determined by the following equation 2.

[equation 2]

$$P_i = \left[ \frac{\alpha_i}{\sum_{k=1}^n} \times (\text{The number of total sub blocks}) \right], \text{ where } i=1 \sim n-1$$

15

$$P_i = (\text{The Number of total sub blocks}) - \sum_{k=1}^{n-1} P_k, \text{ where } i=n$$

Wherein,  $P_i$  is the number of sub block to be requested to each of the nodes,  $\alpha_i$  is connection state valuation index in each of the connections. In other words,  
20 the user client requests sub block of which the number is  $P_i$  to each node.

FIG. 12 is a flow chart of process that is performed when there exists a connection where sub block download is completed according to a preferred

embodiment of the present invention.

Referring to FIG. 12, the download state monitoring module of the connection control module monitors if there exists a connection where sub block download is completed S1200.

5        If there exists a connection where sub block download is completed, a connection of which the download rate is the lowest is searched S1202. The download rate can be calculated by (the number of downloaded sub blocks)/(the number of requested sub blocks).

      When the connection of which the download rate is the lowest is searched,  
10    the number of remaining sub blocks to be downloaded in the connection of which the download rate is the lowest is calculated S1204.

      After the calculating the number remaining sub blocks in the connection of which the download rate is the lowest, it is determined if it is necessary to redistribute sub blocks to be downloaded between the connection where sub block  
15    download is completed and the connection of which the download rate is the lowest S1206. For example, if remaining sub blocks can be downloaded while transmitting control signal for redistributing sub blocks, redistribution of sub blocks would not be necessary.

      If it is determined that redistribution of sub blocks is necessary, the sub  
20    blocks to be downloaded from the connection where sub block download is completed and the connection of which the download rate is the lowest is redistributed according to the connection state valuation index S1208.

      Sub blocks are may be distributed by the following equation 3.

[equation 3]

$$P_1 = \left[ \frac{\alpha_1}{\alpha_1 + \alpha_2} \times (\text{the number of remaining sub blocks}) \right]$$

$$P_2 = (\text{the number of remaining sub blocks}) - P_1$$

5

$P_1$  is the number of sub blocks redistributed to the connection of which the download rate is the lowest,  $\alpha_1$  is the connection state valuation index of the connection of which the download rate is the lowest,  $P_2$  is the number of sub blocks redistributed to the connection where sub block download is completed, and the  $\alpha_2$  is the connection state valuation index of the connection where sub block download is completed.

If sub blocks are redistributed, sub blocks are requested to the node corresponding to each connection according to the new distribution result. If it is not necessary to download sub blocks from the connection of which the sub block download rate is the lowest, the user client requests stoppage of the sub block transmission to the corresponding node.

FIG. 13 is a block diagram of module of the connection control server according to a preferred embodiment of the present invention.

FIG. 13 illustrates module of the connection control server in the system of P2P network of FIG. 12 or in the system multiple servers and P2P are combined such as FIG. 4.

Referring to FIG. 13, the connection control server according to an embodiment of the present invention may comprise a mesh manager module 1300, a node information providing module 1302 and an authentication module 1304. A

user database 1306, a meta information data base 1308 and mesh information database may be coupled to the connection control server.

In FIG. 13, the mesh manager module 1300 manages information on user clients connected to the connection control server. As described above, when the power of the user client is on or the communication agent program is executed, client address information and information of contents stored in the client are transmitted to the connection control server, and the mesh manager module 1300 receives the information and stores it in the mesh information database 1310.

When a user requests specific contents, the node list providing module 1302 derives information of nodes that store the corresponding contents using the mesh information database 1310 and provides the derived information to the user client.

The authentication module 1304 authenticates users and processes billing information. When a user requests contents, authentication key information including user ID and password is transmitted to the connection control server, and the authentication module 1304 determines if the user is a registered user. Further, the authentication module 1304 determines if the user has paid for the contents by the communication with external billing server.

ID of registered users, password, personal information, etc. are stored in the user database 1306, and contents coding method, information of author, etc. is stored in the meta information database 1308. If the user who requests contents is authorized, the mesh manager module 1300 provides meta information to the user. Generally, meta information is necessary for play of the streaming data, however, if play of streaming data is possible without meta information, the meta information database 1308 may not be coupled to the connection control server.

Information of user clients connected to the connection control server and

information of contents stored in each of the user clients are stored in the mesh information database 1310.

FIG. 14 is a block diagram of the module of the connection control server according to another embodiment of the present invention.

5 Referring to FIG. 14, the connection control server may include a server state determining module 1400, a sub block distribution module 1402, a server list providing module 1404 and an authentication module 1406. A user database 1408, contents server database 1410 and meta information database 1412 may be coupled to the connection control server.

10 The server state determining module determines state information of contents servers which are managed by the connection control server. As described above, the server state determining module 1400 uses information including bandwidth, CPU usage rate, memory usage rate, the number of connected users, and file I/O, etc. in order to determine state of each of the contents servers.

15 The server list providing module 1404 provides list information of contents servers which the user client requesting contents should connect to. The server list providing module provides list of servers with good state using the determination result of the server state determining module.

The sub block distribution module 1402 provides information of the number  
20 of sub blocks to be downloaded from each of the contents servers using the state information of the contents servers to the user client which requested contents.

Information of contents servers which the connection control server manages is stored in the contents server database 1410.

Function of the authentication module 1406 and information stored in the  
25 user database 1408 and meta information database 1412 are same as the case of FIG.

13.

### **Industrial Applicability**

5           As described above, according to the present invention, stable streaming service can be provided although error occurs in a connection, because streaming data are provided through multiple paths.

          Further, according to the present invention, download efficiency can be improved by avoiding occurrence of idle connection because sub blocks to be  
10       downloaded are redistributed if sub block download in a connection is completed.

          Especially, the present invention is useful when streaming data are downloaded through the plurality of nodes in P2P network.